DOCUMENT RESUME

ED 300 783 CS 009 391

AUTHOR McConkie, George W.; And Others
TITLE A Microcomputer-Based Software Package for Eye-Monitoring Research. Technical Report No. 434.
INSTITUTION Bolt, Beranek and Newman, Inc., Cambridge, Mass.; Illinois Univ., Urbana. Center for the Study of Reading.
SPONS AGENCY Office of Educational Research and Improvement (ED), Washington, DC.
PUB DATE Aug 88
CONTRACT OEG-0087-C1001
NOTE 23p.
PUB TYPE Reports - Descriptive (141)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Software Reviews; *Data Collection; *Data Processing; *Eye Movements; Eyes; Microcomputers; Research Methodology
IDENTIFIERS Software Evaluation

ABSTRACT

A software package is described that collects and reduces eye behavior data (eye position and pupil size) using an IBM-PC compatible computer. Written in C language for speed and portability, it includes several features: (1) data can be simultaneously collected from other sources (such as electroencephalography and electromyography); (2) logically defined events can be detected in real time on any data channel; and (3) either of two types of data matrix can be produced. Data reduction algorithms and data structures are described. (Five tables of data describing aspects of the software package are included.) (Author/RS)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED300783

CENTER FOR THE STUDY OF READING

A READING RESEARCH AND EDUCATION CENTER REPORT

Technical Report No. 434

A MICROCOMPUTER-BASED SOFTWARE PACKAGE FOR EYE MONITORING RESEARCH

George W. McConkie, Charles W. Scouten,
Patrick K. Bryant, & Jerry H. Wilson

University of Illinois at Urbana-Champaign

August 1988

University of Illinois at Urbana-Champaign
51 Gerty Drive
Champaign, Illinois 61820

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

D. Anderson

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

☐ This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

The work upon which this publication was based was supported in part by the Office of Educational Research and Improvement under Cooperative Agreement No. OEG 0087-C1001. The publication does not necessarily reflect the views of the agency supporting the research.

CS009391

BEST COPY AVAILABLE

Abstract

A software package is described that collects and reduces eye behavior data (eye position and pupil size) using an IBM-PC compatible computer. Written in the C language for speed and portability, it includes several unique features: Data can be simultaneously collected from other sources (i.e., EEG, EMG), logically defined events can be detected in real time on any data channel, and either of two types of data matrix can be produced. Data reduction algorithms and data structures are described.

A MICROCOMPUTER-BASED SOFTWARE PACKAGE FOR EYE MONITORING RESEARCH[1]

The monitoring of eye behavior, including eye position and pupil size data, has proven to be useful in studying issues in a number of different fields, including reading (Rayner, 1983), music reading (Sloboda, 1978), X-ray examination (Nodine & Kundel, 1987), map design (Antes, Chang, & Mullis, 1985), mathematics education (Suppes, Cohen, Laddaga, Anliker, & Floyd, 1982), chess (Chase & Simon, 1973), art and design (Lochegr & Nodine, 1987), mental load (Kahneman & Beatty, 1966), emotional response (Hess, 1972), and oculomotor control (Zuber, 1981). Almost any area of research that involves the acquisition and use of visually-presented information, or in which there are changes in emotional response or arousal, can benefit from the recording of eye position or pupil size.

The upsurge in interest in monitoring the eyes has stimulated the development of new devices for monitoring eye behavior. However, a major factor that has held back a more widespread use of eye monitoring has been the amount of software development required before research is possible. Each laboratory typically writes the software necessary to collect and reduce data with the particular equipment configuration being used.

Recent widespread acceptance of the IBM-initiated standard among personal computers, together with the continually-increasing speed of computers of this family, suggests the appropriateness of developing a set of application programs for this computer family that can be used with a variety of eye behavior monitoring devices. The present paper describes a set of programs that has been developed specifically for the Stoelting Wide Angle Eye Movement Tracker and TV Pupillometer System, but that is compatible with other eye monitoring equipment, as well.

Some Basic Concepts

The Stoelting Tracker/Pupillometer is a device that monitors the eyes by means of a video camera, and contains hardware that processes each video frame to determine the size and location of the pupil. Thus, it samples eye position 60 times per second and gives as output three digital values: 7-bits for horizontal and vertical eye position, and 8-bits for pupil size. In addition, it has a field-of-view camera that allows the stimulus field to be displayed on a monitor, with a fixation spot that moves in response to movements of the eyes. In these respects the equipment is quite similar to video-camera based eye monitoring equipment produced by other companies and is distinguished primarily by a relatively low price.

Each of the different types of eye monitoring devices currently available has strengths and weaknesses (Young & Sheena, 1975). Video-camera based equipment tends to be relatively easy to use, gives an indication of both horizontal and vertical eye position, typically produces pupil size information as well as eye position, and typically provides output showing the direction of gaze superimposed on a television image of the stimulus field. At the same time, most equipment of this type has a relatively slow sampling rate (i.e., 60 samples per second), accuracy of only about 1 deg of visual angle, and a noise level that prevents the reliable detection of small eye movements. Furthermore, there are occasionally artifacts in the signal that result from blinks or failures in the pupil tracking algorithms that may not reflect actual eye movement or pupil size changes. These characteristics make it impossible for equipment of this type to give a precise specificiation of the fine-grained characteristics of eye movements. For example, with samples taken only every 16 msec it is not possible to specify the precise time at which a 30 msec saccadic eye movement began and ended, nor to describe its dynamic characteristics. Also, what appears to be a single eye fixation in the data may actually be a series of eye fixations separated by short saccades. However, since most types of research can be carried out without a higher degree of precision, the equipment is quite adequate for many purposes.

In developing software for video-camera-based equipment, it is necessary to deal with the problems of noise and indeterminacy in the data and to provide the researcher with as much information as possible about the eyes' behavior without falsely interpreting noise and artifacts as eye movements. The effects of the choice of values for parameters in the reduction of noisy eye movement data has been vividly described by Karsh and Breitenbach (1983). The goal of the present development was to deal with the problems posed by video-camera based eye movement data in a way that maximized the valid information available and at the same time would handle appropriately the signal characteristics of other types of eye monitoring devices. These include devices sampling at other rates, and with other noise levels and characteristics. The data collection software has the flexibility to collect data from a variety of Input Ports on the computer, depending upon the hardware configuration of the system. The data reduction software has a number of parameters that allow it to be "tuned" to properly handle eye movement data having different characteristics. Finally, diagnostic information is provided in the data matrix to indicate places in the data where noise characteristics are such that the validity of decisions being made is lower than normal. In this way, the more questionable data can be excluded from use in data analysis if desired.

When eye position is monitored by a tracking device that samples very rapidly (i.e., every msec) and with low noise levels, there is seldom a question about whether a saccade occurred: Indeterminacy in when it began and ended is within a range of a few msec. While the software described here can collect and reduce data in which fixations and saccades are always clearly specified, we will discuss in this article the more difficult case where there is greater indeterminacy. In this latter case, it is not appropriate to speak of fixations and saccades, since these cannot be determined precisely. Rather, we will speak of a *gaze*, which can be defined as the period of time between the onset of one stable period in the eye movement data and the next stable period in which the eyes are directed toward a new location. Each gaze by definition has an initial stable period. The gaze may also have a period at the end that is not stable but that shows a transition toward the next gaze location. The amount of variability in the data that is still classified as "stable," and the amount of change in gaze location that is required in order to reliably detect the fact that the eyes have shifted to a new gaze location, are controlled by parameters in the program.

The Computer Programs

The eye monitoring software package presently consists of four programs: one for data collection, two for data reduction, and one that allows the user to modify calibration values in the data file. They run on IBM compatible personal computers with 256K of memory and a hard disk.

The programs are written primarily in the Microsoft C language using the Microsoft C compiler to achieve both speed and portability. However, this language choice presented certain problems that are discussed below. Flexibility is achieved through parameter files that specify a number of the programs' internal characteristics.

Data Collection Program

The Data Collection Program is designed to collect eye position and pupil size data, as well as data of other kinds, including button presses and analog data of any type. Up to eight 16-bit Input Ports can be recorded. The rate of sampling is determined externally. The program samples on the leading edge of a positive-going pulse that is generated by the eye monitoring device or produced by a clock. The program also allows the user to display stimulus patterns on the computer screen, advancing pages with key or button presses, or to use external displays such as by slide projector or physical placement of photographs in the field of view.

The Data Collection Program is run by giving its name, plus the names of three files: a Parameter File, a Display Control File, and a Data File. The Parameter and Display Control files are ASCII files that can be prepared and modified using any text editor capable of producing pure ASCII output. One

additional file is required containing the hardware initialization code. This file is used to establish communication between the computer and the eye monitoring equipment. For an interface, we currently use the Qu. Tech PXB-721 Parallel Expansion Board, for which A/D modules are also available for monitoring analog signals. If different equipment is used, then this file can be altered to reflect the characteristics of the equipment.

Upon being run, the Data Collection Program requests the subject's name and identification number, and an experimental condition number. All are necessary parameters but shell scripts (i.e., DOS Batch files) can be written that automatically specify some of these parameters. This procedure reduces the likelihood of experimenter error.

The parameters that are contained in the Parameter File are listed in Table 1. These parameters define the hardware environment, the channels on which data are to be collected, and information to be used in the calibration process. This file is read and the parameters invoked as part of the initialization process each time the program is run.

[Insert Table 1 about here.]

The Display Control File defines the sequence of events that are to occur in the experiment. By means of a series of dot commands, shown in Table 2, displays can be put onto the screen, data collection channels can be enabled and disabled, and calibrations can be taken as desired. Alphanumeric display pages can be included in the Display Control File, and can use the IBM extended character set, thus giving some limited graphics capability.

[Insert Table 2 about here.]

The Data Collection Program contains a calibration procedure which, when called in the Internal Display Mode (e.g., using the computer screen to display stimulus material), displays a grid of Xs on the computer screen at locations defined in the Parameter File. The color of each X is modified, in sequence, as a signal for the subject to fixate that point and press a key or button. When the key is pressed, the computer samples the eye data values, saving the values in a calibration table. This is repeated for each location, and the values obtained are compared with the prior values. If eye position values deviate by more than the X or Y Calibration Threshold values specified in the Parameter File, additional samples are taken from that location. This sampling continues until two successive samples have been taken from each location that fall within the threshold values specified or until the experimenter manually exits the calibration procedure. This calibration procedure can be repeated at any time during the data collection process.

When the Data Collection Program is initially called, a nine-point matrix of digits appears on the screen while the experimenter adjusts the eye monitoring equipment. During this time, the values of data channels that are defined as eye position and pupil size are displayed continuously on the bottom of the computer screen. Also, during this time the computer identifies the range of values obtained from the pupil size channel, if available. Thresholds then are calculated for later use in identifying blinks. A Keyboard command next advances the program to the actual data collection procedures, where calibrations and data collection can occur.

In the Data File that is produced by the Data Collection Program, the first 2048 bytes are used to store complete information about the data collection session (see Table 3), including data from each occurrence of the calibration task. Following this, a data record of variable length is created to contain the data collected during each display. This record indicates the channels on which data were collected during that display, an experimental condition identifier for that display, and the actual data which consists of a two-byte integer per sample for each channel.

[Insert Table 3 about here.]

One additional feature of the Data Collection Program is the ability to detect Events and to take action when they occur. An Event is the occurrence of a logical condition in the data coming from a specified data channel. An Event is defined by an identifying title, a channel number, a threshold value, and a logical condition, including equal, not equal, greater than, less than, bitwise AND or bitwise OR. Each defined Event is tested as each data sample is taken. If an Event tests as "true" then immediate action can be taken. Up to four Events can be specified. At present, this feature is used to respond to the press of buttons on one of the data channels, which allows the system to advance the display or to simply record a subject's response. However, this feature provides the basis for a large number of potential response-contingent stimulus manipulations, especially since the Events can be defined on channels other than the eye behavior channels. More detail concerning the techniques used to allow a C language program to deal with real-time Events is presented below.

Data Reduction Programs

Two data reduction programs have been developed. One program is for situations where the primary interest is in eye position data that are of a saccade-fixation type, and the other is for situations where the interest is in non-saccade data, such as pursuit eye movements, pupil size, or EEG. The first program, the Gaze-oriented Reduction Program, produces a matrix in which each data row corresponds to a gaze or fixation-saccade pair. The second program, the Stream Data Reduction Program, is for use when the data cannot appropriately be partitioned into gaze segments. This program produces a row in the data matrix for every data sample taken. However, if eye position data were collected, then the matrix can include columns indicating eye position, the state of the eyes (i.e., stable or in transition), and when gazes begin and end.

Gaze-oriented Data Reduction Program. Table 4 contains definitions and parameters used in the Gaze-oriented Data Reduction Program, and Table 5 presents the general algorithm used in identifying gazes. The algorithm identifies periods of stability in the data, then examines the periods of instability lying between them, attempting to resolve them as either periods of noise in the data or transition periods (i.e., eye movements taking the eyes to a new location). If the period of instability has characteristics that make it inappropriate to interpret in either of these simple ways, then a complex transition analysis is carried out that attempts to break the period into additional gazes using recursion with new parameters. The program produces an ASCII data matrix that contains a row of data for each identified gaze. The columns in the matrix are listed in Table 6. The matrix is in a form typically used with statistical programs and contains information about the location and duration of each gaze, the duration of the stable part of the gaze, and indications of data quality such as the percent of samples with non-aberrant values, amount of drift during the stable part of a gaze (a possible indicator of small, undetected saccades), and abnormal pupil size (an indicator of a blink).

[Insert Tables 4, 5 and 6 about here.]

Once a gaze has been identified, averages of the stable X and Y channel values are calculated. These averages are used to determine where the eyes were directed in the stimulus display during that gaze. The algorithm used in this mapping process produces a solution similar to that of the linear interpolation procedure suggested by McConkie, Wolverton, Zola, and Burns (1978), but in fact accomplishes this goal using recursion.

The mapping algorithm first divides the stimulus space into quadrangles, based on the locations of the grid of target locations used in the calibration task. The algorithm assigns to each calibration target location either the X and Y channel values obtained during the last calibration prior to the display, the first calibration following it, or an average of the values obtained on these two calibrations. The algorithm then takes the average location of the eyes during the stable part of the first gaze and identifies the quadrangle within which it lies. Two lines are then identified that bisect opposing sides of the quadrangle and that divide it into four smaller quadrangles. Again, the algorithm identifies which

of these quadrangles contains the gaze location. This process continues until the gaze location is within a specified distance of one of the corners of a quadrangle, at which point that corner is taken as the gaze location. Since the algorithm knows the corresponding location of that corner in the stimulus field, the mapping of the data point onto a stimulus location is complete and the stimulus-based gaze location is entered into the matrix. The distance threshold used in this process is a parameter in the program. The gaze location, together with the distance and direction of this gaze from prior and following gazes, are entered into the data matrix.

Special provisions are made in the program to handle a case where a gaze location lies outside the region in which calibration targets were used. This is done by extending the nearest quadrangle to include the data point, estimating X and Y values corresponding to the new corners, and then using the normal mapping algorithm to estimate the location of the gaze on the stimulus.

If an Event occurred during a gaze, the matrix also contains information about which Event occurred and the time of its occurrence. This can be used to record response times and/or response selection if each Event defined is the press of a particular button.

The Gaze-oriented Data Reduction program also puts summary information concerning data from non-eye-position channels into the data matrix, including mean value on that channel during the eye fixation, as well as an indication of the direction and amount of change during the gaze period. Finally, the sample number of the first data sample in the gaze is included. This number provides a pointer from the gaze-based matrix into the original data matrix. Thus, it is possible to write programs that select the data collected on some non-eye-movement channel, starting with the onset of a particular gaze or Event.

Stream Data Reduction Program. When the researcher's interest is primarily in changes in pupil size or some other continuous data such as EEG or smooth pursuit eye movement data, the data matrix for analysis is prepared using the Stream Data Reduction Program. This program creates an ASCII matrix having one row for each sample taken. Subject and condition columns are added to the data. If eye position has also been recorded, then columns can be created that indicate for each sample whether it occurred during a period of stability or transition, how many samples have been taken since the beginning of the gaze, where the eyes were directed during that gaze, and whether there was any problem in the eye position data at that time (e.g., blinks or aberrant values). The occurrence of Events can also be noted. With this matrix it is easy to identify the data taken on one channel beginning at the onset of a fixation at a particular location in the display, or beginning at the point where the subject pressed a button or the pupil size exceeded a particular value. Columns in this matrix are listed in Table 7.

[Insert Table 7 about here.]

Calibration Data Editor

A fourth program is included in the software package that allows the experimenter to examine the calibration values contained in a Data File. Furthermore, if it is found that a value is incorrect, then the program allows the experimenter to replace the original value with a new one. This is occasionally necessary when a value obtained in one calibration is very deviant from those obtained during other calibrations in the same session.

Using the C Language for Real-time Programming

The C language that is available on the IBM-PC was not designed for use in real time data collection. In particular, it does not provide a method of responding to most interrupts. Attempting to use it to respond directly to interrupts can actually interfere with the proper execution of some of its standard commands. This fact created certain problems in developing the Data Collection Program, particularly

the need for interrupt driven data collection and the need to respond quickly to the occurrence of an Event. This problem was dealt with by developing an Input Channel Handler written in assembly language. This routine receives commands from the C language program and in turn makes information available to it that can be read upon request.

The Input Channel Handler receives instructions from the calling program concerning which channels to take data from, whether to read 8 or 16 bits of data, the port addresses to be used, where the data are found, the interrupt request line to respond to, and the definitions of Events. When activated, the Handler then responds to each interrupt by sampling the data from each specified Input port and storing it in a buffer. Furthermore, as each sample is taken, that sample is tested against any Event definition specified for that channel. If the test succeeds, then a bit is set in a status word to indicate that the Event occurred.

The C language program can read the 16-bit status word and a 16-bit error word generated by the Input Channel Handler and can request values from the data buffer. For simply obtaining data, the program can request data from the Handler at its own convenience, so long as this occurs frequently enough to avoid overrunning the buffer. However, to respond quickly to Events, it is necessary for the C language program to check the status flag frequently to see if an Event has occurred. By writing the program so that it checks this status at least every msec, it can respond to an Event within 1 msec of its occurrence.

Conclusion

Monitoring eye behavior can play any of three different roles in research. First, it provides data for studying oculomotor control, perception, cognition, emotional response or arousal. Second, eye behavior data can serve as a basis for analyzing simultaneously collected data, allowing selection, elimination or grouping of those data. Third, the eyes can be used to control aspects of the stimulus, such as by changing the display when the eyes are directed at a certain location or sounding a tone when the size of the pupil exceeds a certain value.

The software package described above makes it possible to use an IBM or compatible personal computer to acquire and reduce eye behavior data for any of the three purposes just described. While developed specifically for the Stoelting Wide Angle Eye Movement Tracker and TV Pupillometer System, it is designed to be compatible with a variety of different eye monitoring devices that sample at different rates and have different noise characteristics. The availability of this software should substantially reduce the time and cost involved in establishing the facilities to obtain eye behavior data, thus encouraging its use. One unique contribution of the software is the ability to simultaneously collect data of other types, such as other physiological measures, button presses, or indications of changes taking place in the stimulus field. The second contribution is a general facility for detecting logically-defined Events in the data in real time. This capability can be used for making response-contingent stimulus changes, though this currently requires making user-specific modifications to the existing software. These two characteristics open the door to new and innovative roles for eye monitoring in psychological research.

References

Antes, J. R., Chang, K., & Mullis, C. (1985). The visual effect of map design: An eye-movement analysis. *The American Cartographer, 12,* 143-155.

Chase, W. G., & Simon, H. A. (1973). The mind's eye in chess. In W. G. Chase (Ed.), *Visual information processing* (pp. 215-281). New York: Academic Press.

Hess, E. H. (1972). Pupillometrics: A method of studying mental, emotional, and sensory processes. In N. S. Greenfield & R. A. Sternbach (Eds.), *Handbook of psychophysiology* (pp. 491-531). New York: Holt, Rinehart & Winston.

Kahneman, D., & Beatty, J. (1966). Pupil diameter and load on memory. *Science, 154,* 1583-1585.

Karsh, R., & Breitenbach, F. W. (1983). Looking at looking: The amorphous fixation measure. In R. Groner, C. Menz, D. F. Fisher, & R. A. Monty (Eds.), *Eye movements and psychological functions: International views* (pp. 53-64). Hillsdale, NJ: Erlbaum.

Lochegr, P., & Nodine, C. (1987). Symmetry catches the eye. In J. K. O'Regan & A. Levy-Schoen (Eds.), *Eye movements: From physiology to cognition* (pp. 353-361). Amsterdam: North Holland.

McConkie, G. W., Zola, D., Wolverton, G. S., & Burns, D. D. (1978). Eye movement contingent display control in studying reading. *Behavior Research Methods and Instrumentation, 10,* 154-166.

Nodine, C. F., & Kundel, H. L. (1987). The cognitive side of visual search in radiology. In J. K. O'Regan & A. Levy-Schoen (Eds.), *Eye movements: From physiology to cognition* (pp. 573-583). Amsterdam: North Holland.

Rayner, K. (Ed.) (1983). *Eye movements in reading: Perceptual and language processes*. New York: Academic Press.

Sloboda, J. A. (1978). The psychology of music reading. *Psychology of Music, 6,* 3-20.

Suppes, P., Cohen, M., Laddaga, R., Anliker, J., & Floyd, R. (1982). Research on eye movements in arithmetic performance. In R. Groner & P. Fraisse (Eds.), *Cognition and eye movements* (pp. 57-73). Amsterdam: North Holland.

Young, L. R., & Sheena, D. (1975). Survey of eye movement recording techniques. *Behavior Research Methods and Instrumentation, 7,* 397-429.

Zuber, B. L. (Ed.) (1981). *Models of oculomotor behavior and control*. Boca Raton, FL: CRC Press.

Author Note

Inquiries may be sent to Dr. McConkie at the Center for the Study of Reading, 51 Gerty Drive, Champaign, IL 61820.

Table 1

Data Collection Program Parameters: Parameter File

Number of data collection channels defined.

For each channel (0-7):
- Channel title
- Address (I/O Port)
- Data width (8 or 16 bits)
- Read on interrupt (1=enabled; 0=disabled)

Interrupt request line that is active.

Type of display (Internal or External, where Internal is a display on the computer monitor).

Number of calibration locations.

For each calibration location:
- X value in display space
- Y value in display space
- X location on computer screen
- Y location on computer screen

Calibration threshold on X data.

Calibration threshold on Y data.

Display colors:
- Text color
- Background color
- Emphasized text color

Type of subject interaction during calibration (Keyboard or Event).

Type of subject interaction during data collection (Keyboard or Event).

Table 2

Data Collection Program: Dot Commands in Display Control File

.ep - end of page.
.cb - calibrate.
.nc - no collect (disable all data collection).
.cd - collect data (enable data collection).
.ac - activate a channel for data collection.
.da - deactivate a channel so data no collected.
.ed - event definition.
.eu - event undefined.

Table 3

Data Collection Program: Records in the Data File

System Information Record: 2048 bytes

General Information:		128 bytes
Record type identifier		1 byte = 0
Record size		4 bytes
Program name		64 bytes
Version number: major		1 byte
Version number: minor		1 byte
Date/time		26 bytes
Reserved		31 bytes
Subject Information:		128 bytes
Subject name		64 bytes
Subject ID number		2 bytes
Subject condition code		1 byte
Reserved		61 bytes
File Information:		384 bytes
Parameter File name		64 bytes
Display Control File name		64 bytes
Display condition code		1 byte
Data File name		64 bytes
Reserved		191 bytes
Data Channel Information:		768 bytes
Number of channels (< =8)		1 byte
Channel 0 information		68 bytes
Channel Title	64 bytes	
I/O Port	2 bytes	
Data width (8/16)	1 byte	
Read on interrupt flag (1=true)	1 byte	
Channel 1 information		68 bytes
. . .		
Channel 7 information		68 bytes
Reserved		223 bytes
Data Collection Information:		128 bytes
Data collection rate (samples per second)		2 bytes
Number of ASCII characters used to define page condition		1 byte
Reserved		125 bytes
Reserved:		512 bytes

Table 3 (Continued)

Calibration Record:	1024 bytes	
Record type identifier		1 byte = 1
Size of record		2 bytes
Number of calibration points		1 byte
For each calibration point:		
Value assigned to X location		2 bytes
Value assigned to Y location		2 bytes
For each calibration:		
Preceding display number		1 byte
For each calibration point in each calibration:		
Obtained X value		2 bytes
Obtained Y value		2 bytes
Obtained Pupil size		2 bytes
Status of next calibration (0=good, 1=aborted, -1=nonexistent)		1 byte
Event Record:	384 bytes	
Record type identifier		1 byte = 3
Record size		4 bytes
Number of Events defined (<=4)		1 byte
Definition of Event 0		68 bytes
Event title	64 bytes	
Channel number	1 byte	
Threshold value	2 bytes	
Logical condition	1 byte	
0 = equal		
1 = not equal		
2 = greater than		
3 = less than		
4 = bitwise AND		
5 = bitwise OR		
Definition of Event 1		68 bytes
Definition of Event 2		68 bytes
Definition of Event 3		68 bytes
Data Record:	(length unspecified)	
Record type identifier		1 byte = 2
Record size		4 bytes
Display number		1 byte
Channel mask (Channels on which data are collected)		1 byte
Page condition		80 bytes
Data (per channel per sample)		2 bytes

Table 4

Gaze-oriented Data Reduction Program: Definitions and Parameters

Term or Parameter (Parameters are underlined)	Definition
X Channel:	The data channel that gives information about horizontal eye position.
Y Channel:	The data channel that gives information about vertical eye position.
Gaze:	The eye movement record is used to segment time into discrete periods, each termed a Gaze. Each Gaze begins when the eyes stabilize in a new position.
Stable Period:	The initial period of a Gaze, during which the eyes are relatively still.
Gaze Location:	The location in display space to which the eyes are directed during a Stable Period.
Gaze Duration:	The number of samples included in a Stable Period.
Transition Period:	The final period of a Gaze, during which the eyes move to a new location.
Transition Length:	The distance between the Gaze Locations that precede and follow the Transition Period, measured in units appropriate to the stimulus display.
Transition Direction:	The angle of the line connecting two successive Gaze Locations, measured in degrees, where zero deg is a horizontal line extending directly rightward from a Gaze Location, locations above that location have positive values, and locations below have negative values.
Transition Duration:	The number of samples included in a Transition Period.
Local Gaze:	A specified number of samples in which the maximum and minimum values on both the X and the Y channels differ by no more than the values defined as thresholds for those channels.
Local Gaze Duration:	The number of samples in a Local Gaze. (5)
Local Gaze Threshold X:	The maximum difference that can exist among a set of X data values and still be considered a Local Gaze. (1)

Table 4 (Continued)

Term or Parameter (Parameters are underlined)	Definition
Local Gaze Threshold Y:	The maximum difference that can exist among a set of Y data values and still be considered a Local Gaze. (1)
Transition Data:	A string of data values that do not contain a region that satisfies the definition of a Local Gaze. Thus, a candidate for containing a Transition Period.
Consistent Transition Data:	Transition Data values from one channel that change in only one direction.
Inconsistent Transition Data:	Transition Data values from one channel that both increase and decrease.
Reduced Local Gaze:	A specified number of samples, fewer than the Local Gaze Duration, that meet the Local Gaze Threshold Definition for a Local Gaze. The Reduced Local Gaze is used in Complex Transition Analysis.
Reduced Local Gaze Duration:	The number of samples in a Reduced Local Gaze. (3)
Aberrant Value:	A data value from either the X or Y channel that differs from both the preceding and following data values by at least a specified amount, and in the same direction. Thus, it is a local peak or valley in the data stream, and is assumed to represent noise rather than actual change in eye position.
Aberrant Value Threshold:	The threshold that defines an aberrant data value. (2)
Transition Analysis:	A procedure for examining Transition Data in order to determine whether or not a Transition Period exists.
Minimum Gaze Duration:	The minimum number of samples that can exist between the initial samples of two successive Local Gazes, without collapsing these into a single Gaze. Notice that this parameter is used in dealing with the case of overlapping Local Gazes. (3)
Successive Gaze Thresholds - X & Y:	If the locations of two successive gazes differ by less than the Successive Gaze Thresholds on both X and Y channels, they are collapsed into a single Gaze. (1)

Table 4 (Continued)

Term or Parameter (Parameters are underlined)	Definition
Complex Transition Period:	Transition Data that (a) contain more samples than specified by Transition Duration Threshold, or (b) have fewer samples than this threshold, are inconsistent (see definition above), and with preceding and following Gaze Locations that differ by more than the Successive Gaze Threshold. Transition Data not meeting either criterion are dealt with in the normal Transition Analysis.
Transition Duration Threshold:	A threshold used in determining whether a set of Transition Data constitutes a simple or complex transition period. (3)
Complex Transition Analysis:	A procedure for further analyzing a Complex Transition Period to determine whether it can be broken up into additional Gazes. This procedure uses the Reduced Local Gaze in its attempt to find Stable Periods.

Note: Default values for parameters used with Stoelting Pupillometer are given in parentheses.

Table 5

Gaze-oriented Data Reduction Program: Procedure

Given: A Local Gaze has been identified.

A. Find the next series of data values that qualifies as a Local Gaze.

1. If the new Local Gaze begins one data value after the beginning of the prior Local Gaze, extend the current Gaze to include new data point. Go to A.

2. If the new Local Gaze begins no more than a MINIMUM GAZE DURATION of values following the beginning of the prior Local Gaze, extend the current Gaze to include these data. Go to A. [Note: This includes cases where Local Gazes overlap substantially but are not sequential, indicating the possibility of some movement during the gaze, but too little to identify as saccades.]

3. Else, carry out Transition Analysis to examine the Transition Data.

   a) If the difference between either the average X or Y data values from the preceding and following stable periods exceeds the GAZE MOVEMENT THRESHOLD, then set the Movement Flag = TRUE else set it FALSE. [Set a flag if the eyes have changed location.]

   b) Set the Transition Duration Code to indicate the duration of the Transition Period. If Transition Duration (TD) = 0, then set Transition Duration Code (TDC) = 0; if TD = 1 then set TDC = 1; if TD is greater than 1 but less than the Transition Duration Threshold then set TDC = 2; if TD is greater than or equal to the Transition Duration Threshold then set TD = 3.

   c) If the Transition Data from either the X or the Y channel are Inconsistent, then set the Consistency Flag = FALSE else set it TRUE. [Use a flag to indicate whether or not the data show the eyes proceeding consistently in one direction.]

   d) If the Movement Flag = TRUE and the Consistency Flag = TRUE and the Transition Duration Code < 3 then classify this as a clean transition, close out the old Gaze, and go to A. [A short, consistent transition that takes the eyes to a new location is accepted as a saccadic movement.]

   e) If the Transition Duration Code = 3 or if the Movement Flag = TRUE and the Transition Duration Code = 2 and the Consistency Flag = FALSE then go to 4 and carry out Complex Transition Analysis. [If this is a long transition or if the direction of movement is not consistent, it needs additional analysis.]

   f) If the Transition Duration Code < 3 and the Movement Flag = FALSE and the Consistency Flag = FALSE, then mark all Aberrant Values, combine the preceding and following Stable Periods into one Gaze, and go to A. [If this is a period of inconsistent movement with the eyes ending up in the same place, conclude that it is just a period of noise and include it as part of the gaze.]

Table 5 (Continued)

g) If the Transition Duration Code < 3 and the Movement Flag = FALSE and the Consistency Flag = TRUE, then combine the preceding and following Stable Periods into one Gaze, and go to A. [This is a condition in which there is a period of consistent movement that leaves the eyes in the same location. It is included for completeness, but should not normally occur.]

h) If the Transition Duration Code < 3 and the Movement Flag = TRUE and the Consistency Flag = FALSE, then mark the Aberrant Values. If an Aberrant Value is adjacent to a Stable Period, then join it to that Period, removing it from the Transition Period. The remaining Transition Period is accepted as a legitimate transition. Go to A. [If the eyes have moved, but the transition period is noisy, extend the Stable Period if possible and accept the transition.]

B. Carry out Complex Transition Analysis:

1. Mark all Aberrant Values in Transition Data. Consider them to be equal to their bounding values, and reapply Transition Analysis. If this eliminates all Transition Data, go to A. [The transition was apparently just an extended noisy period.]

2. Else use Reduced Local Gaze definition and reapply Transition Analysis to remaining Transition Data. Indicate in the Problem Flag when data show unusual characteristics. Go to A. [This call will find shorter stable periods or periods during which there is more drift or movement, but which do not show the characteristics of saccadic movements. Thus, it has the capability of decomposing a long string of Transition Data into additional gazes.]

Table 6

Gaze-oriented Data Reduction Program: Data Matrix

Col.	Bytes	Contents
1.	4	Subject number
2.	2	Subject condition code
3.	3	Display number
4.	As defined	Display condition code
5.	3	Gaze number on display
6.	6	Eye position on display: X
7.	6	Eye position on display: Y
8.	3	Percent of values that are non-aberrant: X
9.	3	Percent of values that are non-aberrant: Y
10.	5	Distance from prior Gaze
11.	6	Direction from prior Gaze (degrees)
12.	5	Distance to following Gaze
13.	6	Direction to following Gaze (degrees)
14.	4	Total number of samples in Gaze
15.	4	Number of Samples in stable part of Gaze
16.	7	Problem Flag
17.	5	Sample number of first sample in Gaze
18.	4	Number of aberrant samples
19.	5	Average X value in Gaze
20.	5	Average Y value in Gaze
21.	4	Change in X during Gaze
22.	4	Change in Y during Gaze
	For each Event, when it occurs:	
E1.	2	Event number
E2.	4	Time since beginning of Gaze
E3.	5	Time since beginning of Display
	For each Data Channel other than X and Y Eye Position:	
D1.	5	Mean data value during Gaze
D2.	4	Maximum data value during Gaze
D3.	4	Minimum data value during Gaze
D4.	4	Mode data value during Gaze
D5.	5	Slope of data during Gaze

Table 7

Stream Data Reduction Program: Data Matrix

Col.	Bytes	Contents
1.	4	Subject number
2.	2	Subject condition code
3.	3	Display number
4.	As defined	Display condition code
5.	5	Data sample number
	For Each Channel (0-7):	
C1.	5	Channel data value
	Gaze Information:	
G1.	3	Gaze number on display
G2.	6	Eye position on display: X
G3.	6	Eye position on display: Y
G4.	4	Total number of samples in Gaze
G5.	4	Number of samples in stable part of Gaze
G6.	4	Sample number in the Gaze
G7.	7	Problem flag for Gaze
G8.	4	Percent of samples that are aberrant
G9.	3	Blink flag
G10.	2	Aberrant value flag: X
G11.	2	Aberrant value flag: Y
	For Each Defined Event (0-3):	
E1.	2	Event type
E2.	4	Number of samples since Gaze beginning
E3.	5	Number of samples since display beginning

EDITORIAL ADVISORY BOARD
1988-89

Beck, Diana

Commeyras, Michelle

Foertsch, Daniel

Hartman, Doug

Jacobson, Michael

Jihn-Chang, Jehng

Jimenez, Robert

Kerr, Bonnie

Kerr, Paul

Meyer, Jennifer

Moran, Juan

Ohtsuka, Keisuke

Roe, Mary

Schommer, Marlo

Scott, Judy

Stallman, Anne

Wilkinson, Ian

Wolff, Phillip

MANAGING EDITOR
Mary A. Foertsch

MANUSCRIPT PRODUCTION ASSISTANTS
Delores Plowman
Nancy Diedrich